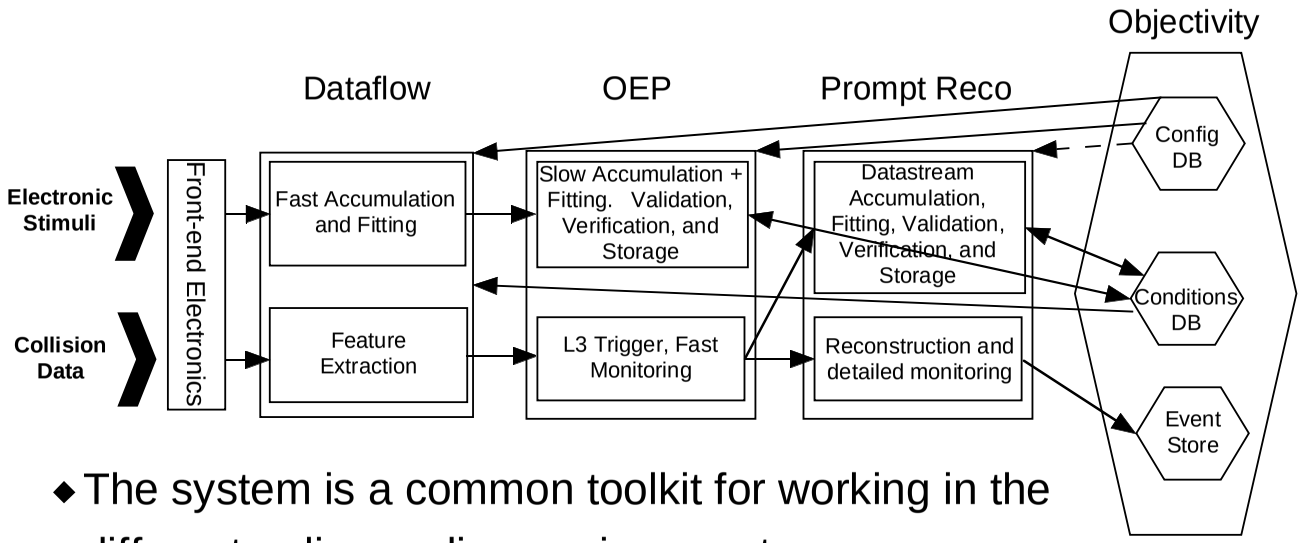
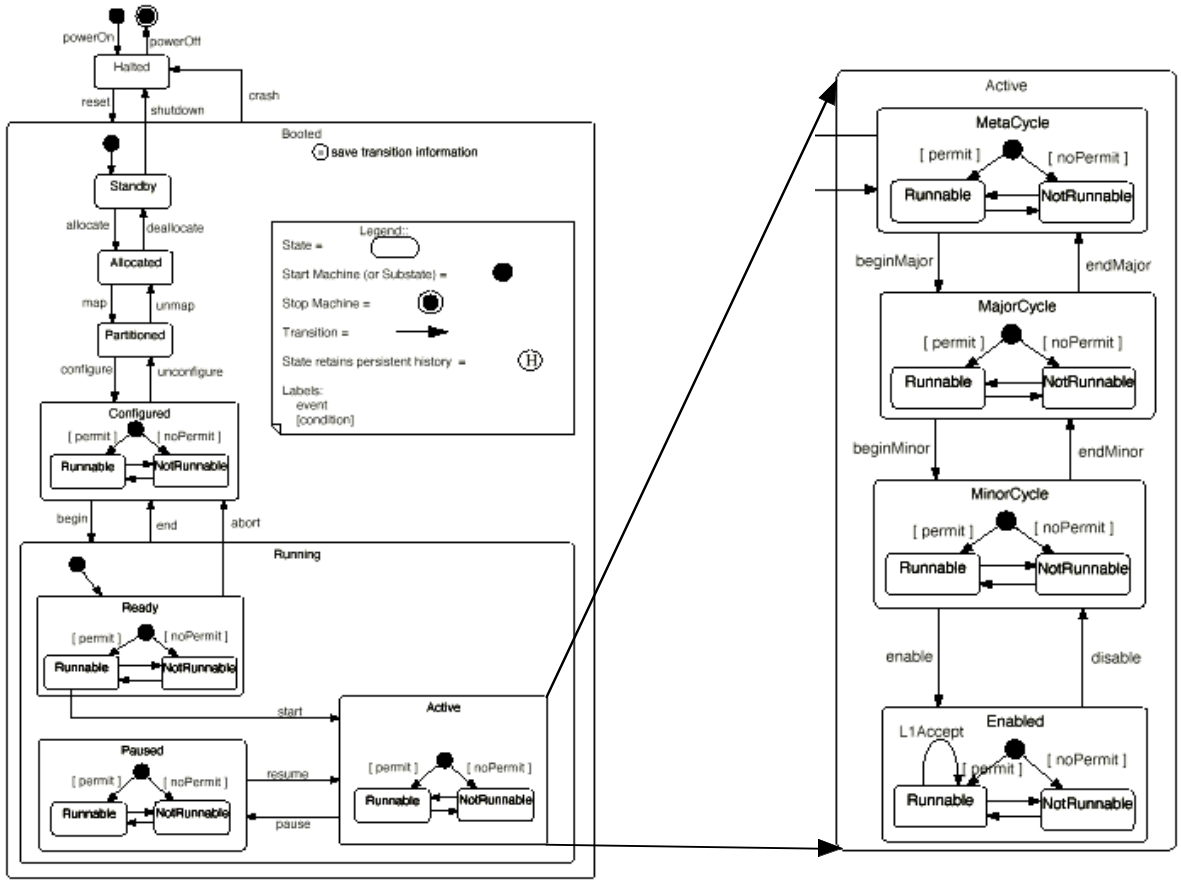
BaBar Calibration System

- One system is used both for online (electronics) and offline (datastream) calibration.



- The system is a common toolkit for working in the different online coding environments
- Accumulation, Fitting, Validation, Verification and Storage are defined in a coherent way

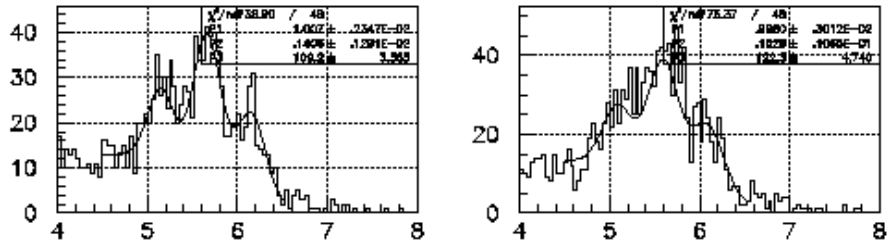
FSM State Diagram



Calibration Progress and Status

- Accumulation
  - Support for nested calibration cycles (EG EMC)
- Fitting (Matt Weaver, CIT)
  - Sophisticated tools for Gaussian, Nonlinear, Maximum Likelihood fitting
  - Tools for comparing histograms (Kolmogorov-Smirnov)
  - Works in all online environements (Dataflow, OEP, PR)

Simulated EMC source spectra + fits



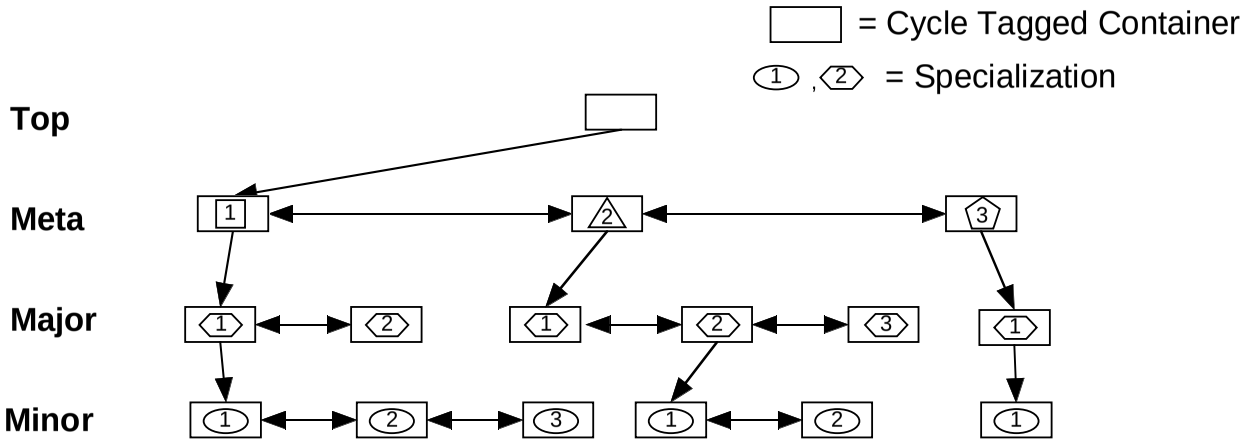
- Conditions Database Interface
  - Stable and functional for both online and offline (read) access
  - Performance problems understood, fix understood

Calibration Progress and Status (cont.)

- Interactive Access
  - New classes/structure for converting calibration objects to viewable (HepTuple) objects (Vasilia Shelkov, LBL)
    - Collection of channel calibration constants becomes an Ntuple, 1 row/channel, 1 column/field
    - CalHistChan becomes HepHistogram
  - Progress on calibration DB browser (Alex Romosan, LBL)
    - Better OO design of Motif interface
    - Generalization of browser interface to other databases (IE config)
- Dataflow Integration
  - Port to unix side of dataflow nearly done
    - Tests online package dependencies
  - Port to VxWorks to begin soon
- OEP Integration
  - Calibration Tagged Container → CalList converter exists
- Prompt Reco Integration
  - Prototype design of distributed object and interface (Objectivity)
  - Not needed for MDC2/V3 release (existing code works up to cosmic run)

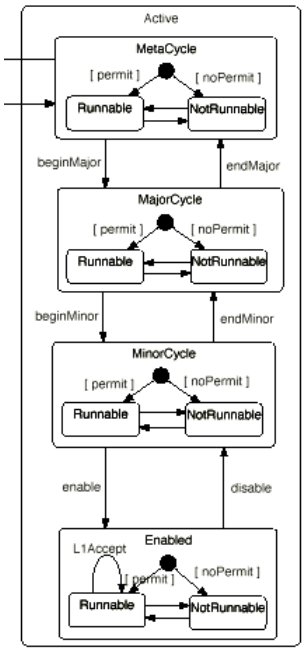
Calibration Progress and Status (cont.)

- Sequencing
  - General scheme for consistent sequencing designed + implemented
    - (distributed) Tagged Container Heirarchy describes cycle steps
    - Specialization through subclassing (IE DAC values)
  - Instantiation of test objects from Config DB (Yury Kolomensky, CIT)
    - Persistent object is transient (TC heirarchy) factory
    - Specialization through subclassing
  - Simple download mechanism understood
    - User interface under design



Sequencing the Dataflow State Machine

- FSM can be sequenced by browsing the CycleTC heirarchy
  - Integration with RunControl is in progress



CalCycleManager output (subclass of odfManger)

```
CalCycleManager: Allocate transition completed with damage = 0
CalCycleManager: Map transition completed with damage = 0

Enter the configure environment value (in hex): f

CalCycleManager: Configure transition completed with damage = 0
CalCycleManager: Begin transition completed with damage = 0
CalCycleManager: Start transition completed with damage = 0
CalCycleManager: BeginMajor transition completed with damage = 0
CalCycleManager: BeginMinor transition completed with damage = 0
CalCycleManager: Enable transition completed with damage = 0
CalCycleManager: Disable transition completed with damage = 0
CalCycleManager: EndMinor transition completed with damage = 0
CalCycleManager: BeginMinor transition completed with damage = 0
CalCycleManager: Enable transition completed with damage = 0
CalCycleManager: Disable transition completed with damage = 0
CalCycleManager: EndMinor transition completed with damage = 0
CalCycleManager: EndMajor transition completed with damage = 0
CalCycleManager: End transition completed with damage = 0
```

Still Missing for V3 Online Release

- Example OEP slow calibration module
- Example odfAction set for fast calibration
- Download (good interface, simple implementation)
- Put it all together